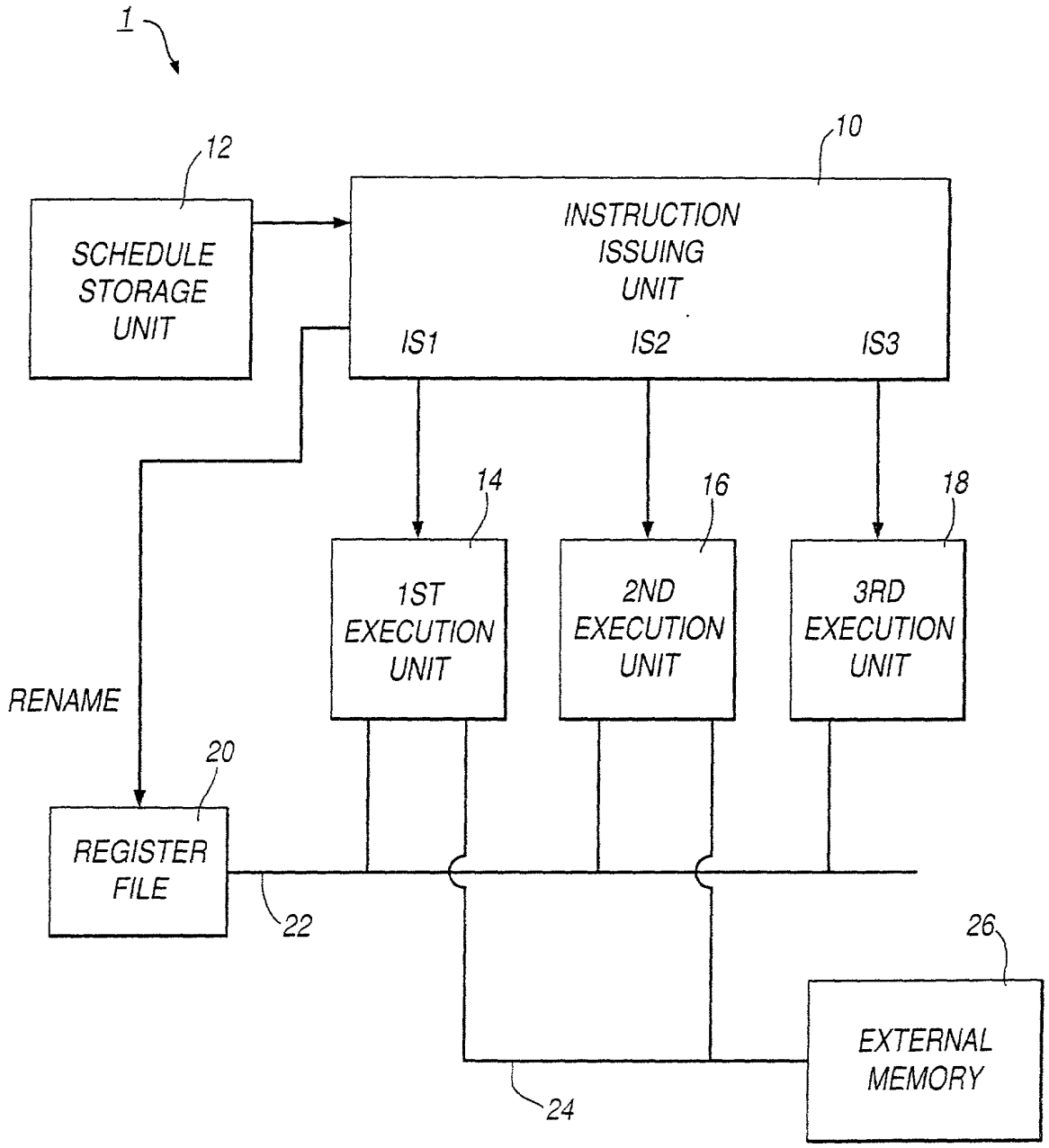


Fig.1

09/27/05 09:09:01

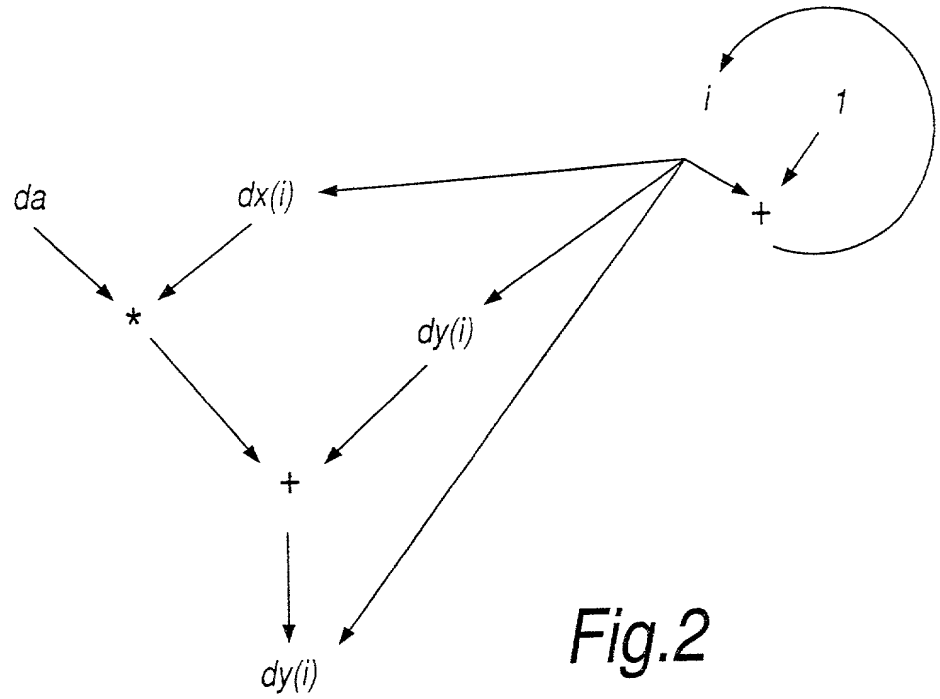

i
1
+
da
dx(i)
*
dy(i)
+
dy(i)

Fig.2

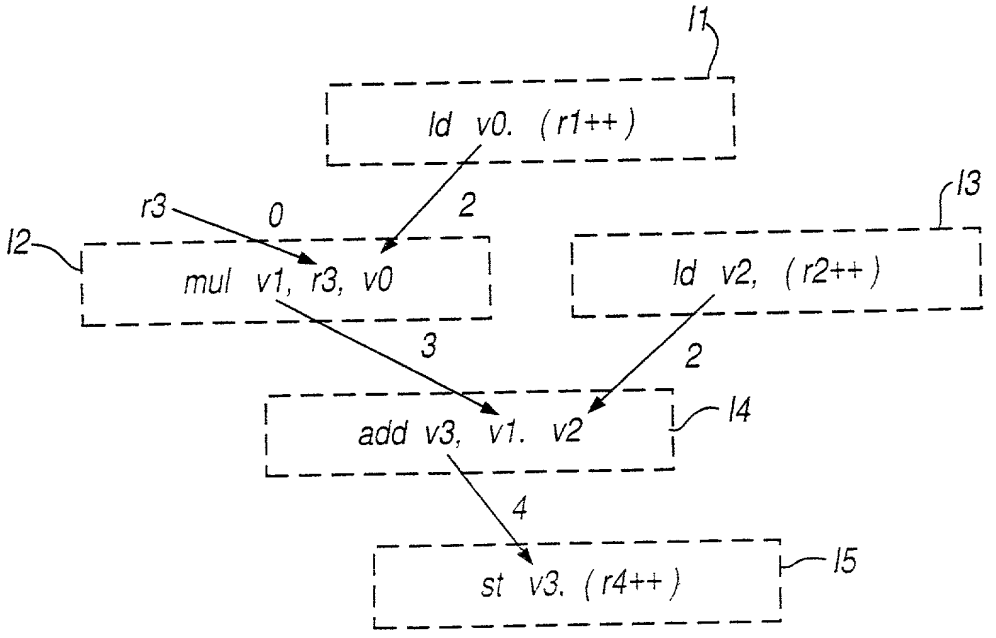

I1
ld v0. (r1++)
r3
0
2
I2
mul v1, r3, v0
I3
ld v2, (r2++)
3
2
add v3, v1. v2
I4
4
st v3. (r4++)
I5

Fig.3

Cycle	i=0	i=1	i=2	i=3	i=4	i=5
0	L1					
1						
2	M	L1				
3	L2					
4		M	L1			
5	A	L2				
6			M	L1		
7		A	L2			
8				M	L1	
9	S		A	L2		
10					M	L1
11		S		A	L2	
12						M
13			S		A	L2
14						
15				S		A
16						
17					S	
18						
19						S

Fig.4

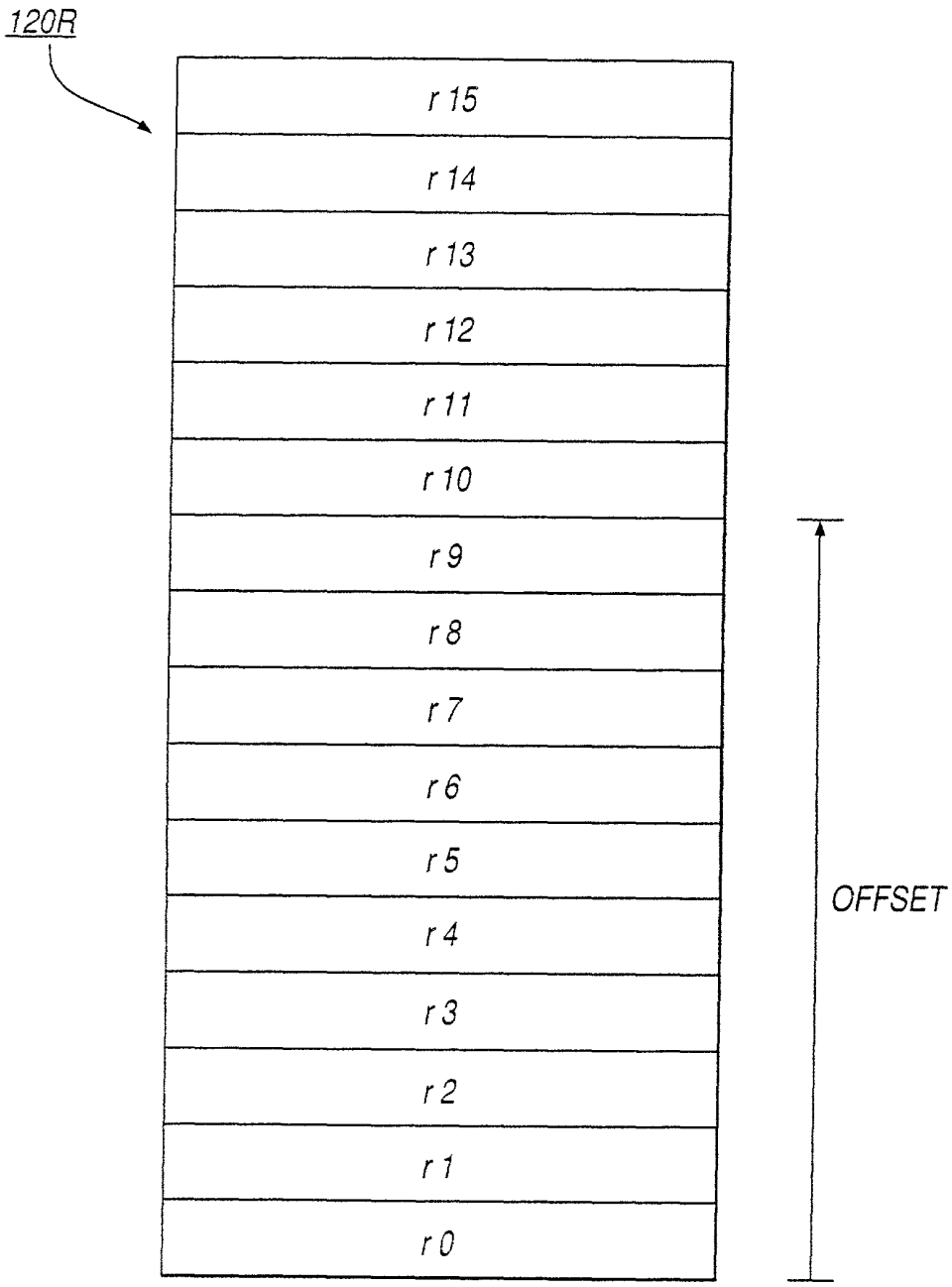


Fig.5

09/29/2003, EAST Version: 1.03.0002

COMMUNICATING INSTRUCTION RESULTS IN. . .
Nigel Peter Topham
Greer, Burns & Crain, Ltd. (Patrick Burns)
Ref. No.0808.65202
Sheet 5 of 10
(312) 360 0080

		Iteration 0					Iteration 1				
Cycle	Offset	Instruction	v0	v1	v2	v3	Instruction	v0	v1	v2	v3
0	10	ld v0, (r1++)	s0 ▸ r10								
1	10		s0 ▸ r10								
2	9	mul v1, v0, r3	s1 ▸ r10	s2 ▸ r11			ld v0, (r1 ++)	s0 ▸ r9			
3	9	ld v2, (r2++)		s2 ▸ r11	s4 ▸ r13			s0 ▸ r9			
4	8			s3 ▸ r11	s5 ▸ r13		mul v1, v0, r3	s1 ▸ r9	s2 ▸ r10		
5	8	add v3, v1, v2		s3 ▸ r11	s5 ▸ r13	s6 ▸ r14	ld v2, (r2++)		s2 ▸ r10	s4 ▸ r12	
6	7					s7 ▸ r14			s3 ▸ r10	s5 ▸ r12	
7	7					s7 ▸ r14	add v3, v1, v2		s3 ▸ r10	s5 ▸ r12	s6 ▸ r13
8	6					s8 ▸ r14					s7 ▸ r13
9	6	st v3, (r4++)				s8 ▸ r14					s7 ▸ r13
10	5										s8 ▸ r13
11	5						st v3, (r4++)				s8 ▸ r13

Translation:

I 1 ld v0, (r1++) ⟶ ld s0, (r1++)

I 2 mul v1, v0, r3 ⟶ mul s2, s1, r3

I 3 ld v2, (r2++) ⟶ ld s4, (r2++)

I 4 add v3, v1, v2 ⟶ add s6, s3, s5

I 5 st v3, (r4++) ⟶ st s8, (r4++)

Fig.6(A)

COMMUNICATING INSTRUCTION RESULTS IN. . .
Nigel Peter Topham
Greer, Burns & Crain, Ltd. (Patrick Burns)
Ref. NC 0808.65202
Sheet 6 of 10
(312) 360 0080

		Iteration 2					Iteration 3				
Cycle	Offset	Instruction	v0	v1	v2	v3	Instruction	v0	v1	v2	v3
0	10										
1	10										
2	9										
3	9										
4	8	ld v0, (r1++)	s0 ▸ r8								
5	8		s0 ▸ r8								
6	7	mul v1, v0, r3	s1 ▸ r8	s2 ▸ r9			ld v0, (r1 ++)	s0 ▸ r7			
7	7	ld v2, (r2++)		s2 ▸ r9	s4 ▸ r11			s0 ▸ r7			
8	6			s3 ▸ r9	s5 ▸ r11		mul v1, v0, r3	s1 ▸ r7	s2 ▸ r8		
9	6	add v3, v1, v2		s3 ▸ r9	s5 ▸ r11	s6 ▸ r12	ld v2, (r2++)		s2 ▸ r8	s4 ▸ r10	
10	5					s7 ▸ r12			s3 ▸ r8	s5 ▸ r10	
11	5					s7 ▸ r12	add v3, v1, v2		s3 ▸ r8	s5 ▸ r10	s6 ▸ r11
12	4					s8 ▸ r12					s7 ▸ r11
13	4	st v3, (r4++)				s8 ▸ r12					s7 ▸ r11
14	3										s8 ▸ r11
15	3						st v3, (r4++)				s8 ▸ r11

Fig.6(B)

Cycle	i=0	i=1	i=2	i=3	i=4	i=5
0	L1					
1						
2	M	L1				
3	L2					
4		M	L1			
5	A	L2				
6			M	L1		
7		A	L2			
8				M (1)	L1 (0)	
9	S		A (3)	L2 (2)		
10					M (5)	L1 (4)
11		S		A (7)	L2 (6)	
12						M (9)
13			S		A (11)	L2 (10)
14						
15				S		A (15)
16						
17					S	
18						
19						S

Fig.7

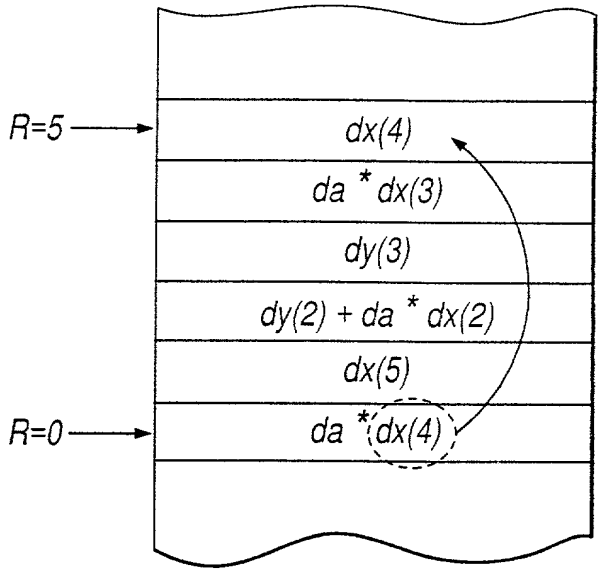

R=5
dx(4)
da * dx(3)
dy(3)
dy(2) + da * dx(2)
dx(5)
R=0
da * dx(4)

Fig.8

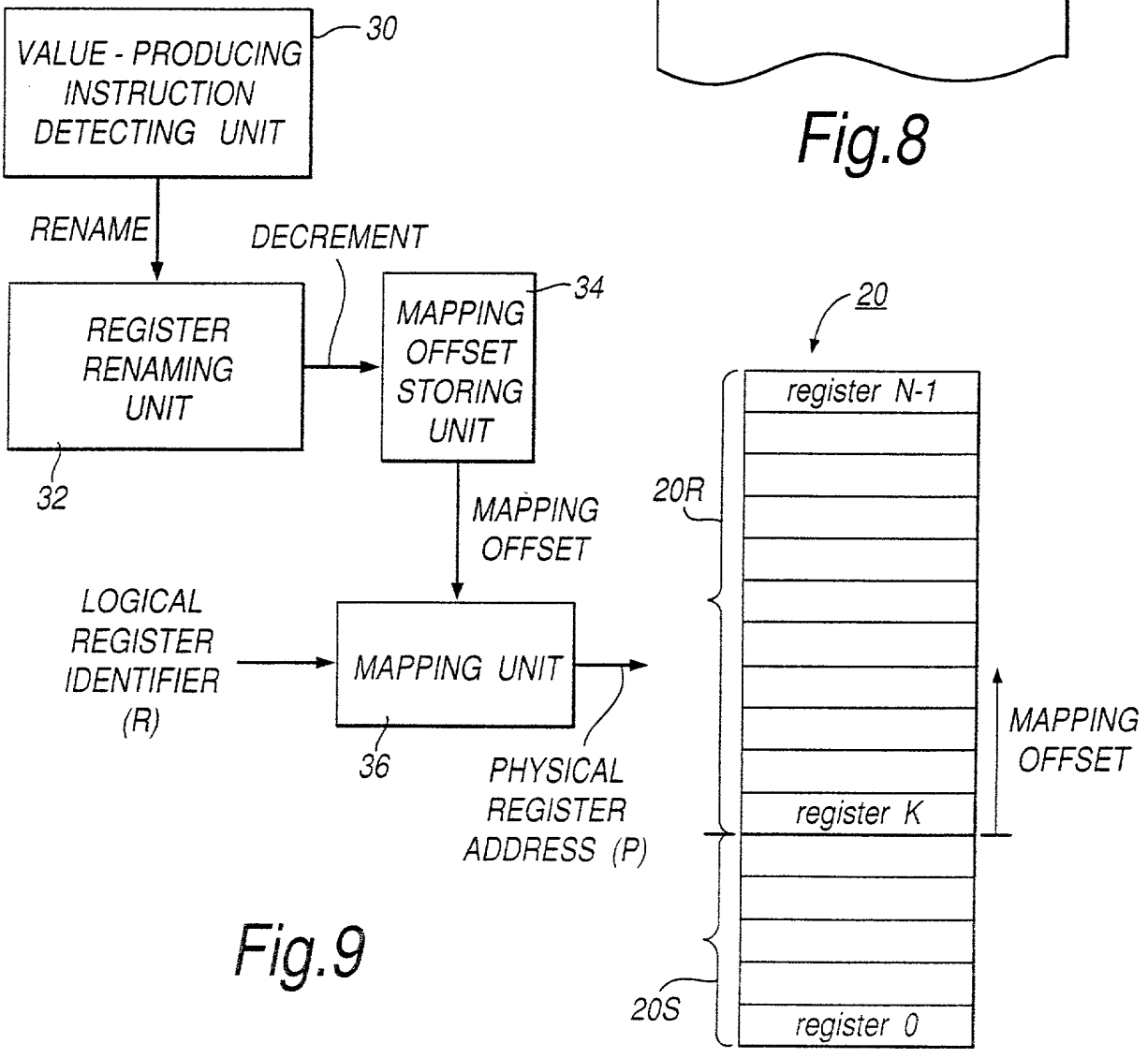

VALUE - PRODUCING INSTRUCTION DETECTING UNIT
30
RENAME
DECREMENT
REGISTER RENAMING UNIT
32
MAPPING OFFSET STORING UNIT
34
MAPPING OFFSET
LOGICAL REGISTER IDENTIFIER (R)
MAPPING UNIT
36
PHYSICAL REGISTER ADDRESS (P)
20
register N-1
20R
register K
MAPPING OFFSET
20S
register 0

Fig.9

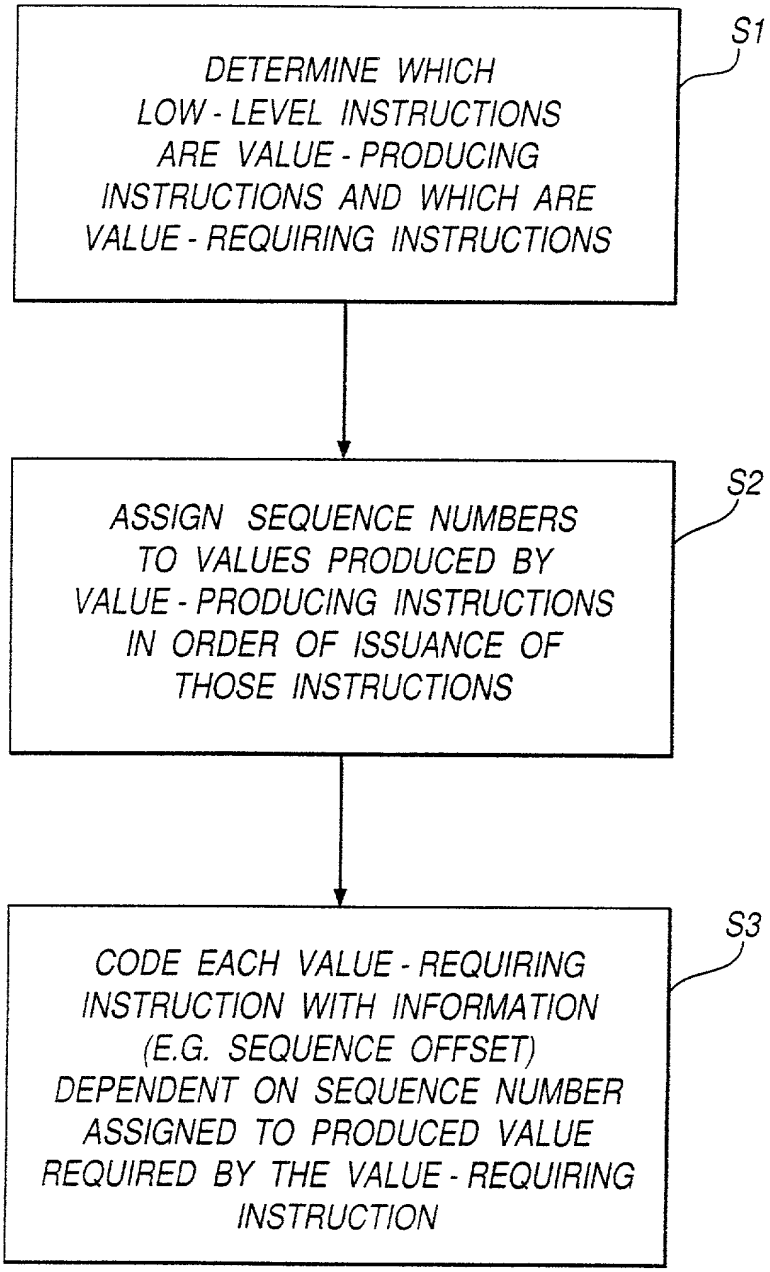

DETERMINE WHICH LOW - LEVEL INSTRUCTIONS ARE VALUE - PRODUCING INSTRUCTIONS AND WHICH ARE VALUE - REQUIRING INSTRUCTIONS
S1
ASSIGN SEQUENCE NUMBERS TO VALUES PRODUCED BY VALUE - PRODUCING INSTRUCTIONS IN ORDER OF ISSUANCE OF THOSE INSTRUCTIONS
S2
CODE EACH VALUE - REQUIRING INSTRUCTION WITH INFORMATION (E.G. SEQUENCE OFFSET) DEPENDENT ON SEQUENCE NUMBER ASSIGNED TO PRODUCED VALUE REQUIRED BY THE VALUE - REQUIRING INSTRUCTION
S3

Fig.10

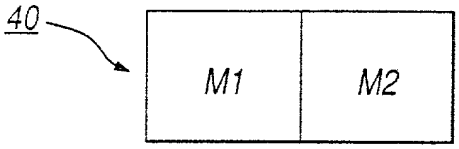


Fig.11

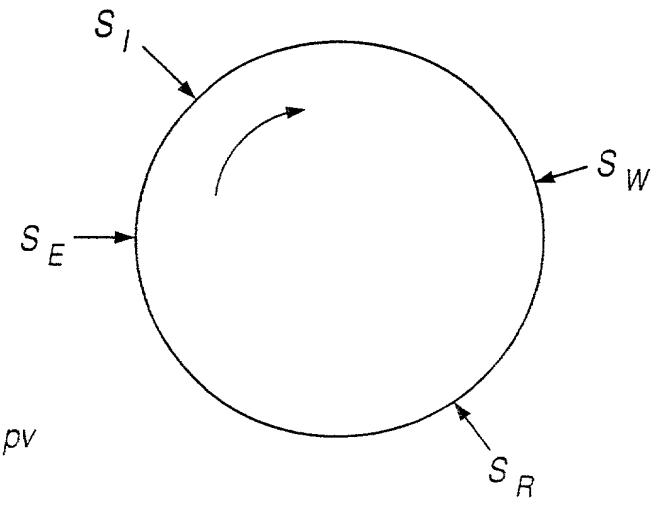
```
1:    int s = 0;
2:    for ( i = 0; i < N; i++ )
3:    {
4:          s = s + a[i];
5:          b[i] = s;
6:    }
7:    c = s;
```

Fig.12

$S_I - S_R = q$

$S_W - S_E = t$

$S_W - S_R = v$



circumference = pv

Fig.13